AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (original): A computer implemented method of load balancing a multiprocessor computer system, comprising the following steps:

A2 determining the state of each of two or more processors, wherein the state includes at least one of a source and sink state; and

if at least one of the two or more processors is in a source state and at least one of the two or more processors is in a sink state, transferring at least one thread from a queue of a source state processor to a queue of a sink state processor.

2. (original): The method as claimed in claim 1, wherein the state further includes a neither state.

3. (original): The method as claimed in claim 1, wherein the method further comprises the following step:

repeating said steps.

4. (original): The method as claimed in claim 1, wherein the method is initiated once every second.

5. (original): The method as claimed in claim 1, wherein the method is performed indefinitely.

6. (original): The method as claimed in claim 1, wherein the method further includes the following step:

determining the load of each of the two or more processors.

7. (original): The method as claimed in claim 6, wherein the transferring step further includes:

transferring at least one thread from the highest loaded, source state processor to the lowest loaded, sink state processor.

8. (currently amended): A computer implemented method of load balancing a multiprocessor computer system, comprising the following steps:

determining a score of each of two or more processors wherein the score is a function of at least a processor state;

AZ

determining a best score processor and a worst score processor; and

transferring at least one thread from a queue of a worst score processor to a queue of a best score processor.

9. (canceled)

10. (original): The method as claimed in claim 8, wherein the score is a function of at least a processor state and a processor load.

11. (original): The method as claimed in claim 10, wherein the processor state is weighted more heavily than the processor load.

12. (original): A computer implemented method of load balancing a networked plurality of computer systems, comprising the following steps:

determining the state of each of the networked plurality of computer systems, wherein the state includes at least one of a source and sink state; and

if at least one of the plurality of computer systems is in a source state and at least one of the plurality of computer systems is in a sink state, transferring at least one thread from a source state processor to a sink state processor.

13. (original): A computer system for balancing load using starvation avoidance comprising:

one or more processors for receiving and transmitting data; and

a memory coupled to said one or more processors, said memory having stored therein sequences of instructions which, when executed by one of said one or more processors, cause one of said one or more processors to determine the state of each of said one or more processors, wherein the state includes at least one of a source and sink state, and, if at least one of the one or more processors is in a source state and at least one of the one or more processors is in a sink state, transfer at least one thread from a source state processor to a sink state processor.

AJ

14. (new): The method as claimed in claim 12, wherein the state further includes a neither state.

15. (new): The method as claimed in claim 12, wherein the method further includes the following step:

determining the load of each of the computer systems.

16. (new): The method as claimed in claim 15, wherein the transferring step further includes:

transferring at least one thread from the highest loaded, source state computer system to the lowest loaded, sink state computer system.

17. (new): The system as claimed in claim 13, wherein the state further includes a neither state.

18. (new): The system as claimed in claim 13, wherein the system further includes instructions which, when executed by one of said one or more processors, cause one of said one or more processors to determine the load of each of the one or more processors.

19. (new): The system as claimed in claim 18, wherein the instructions causing the processor to transfer at least one thread from a source state processor to a sink state processor further include instructions which, when executed by one or said one or more processors, cause one of said one or more processors to transfer at least one thread from the highest loaded, source state processor to the lowest loaded, sink state processor.